UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/742,745	12/20/2000	Charles P. Roth	10559/432001/ P10502-AD1	6467

20985 7590 11/05/2004

FISH & RICHARDSON, PC
12390 EL CAMINO REAL
SAN DIEGO, CA 92130-2081

EXAMINER	
GERSTL, SHANE F	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 11/05/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/742,745	ROTH ET AL.	
	Examiner	Art Unit	
	Shane F Gerstl	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *8/12/04*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-11 and 13-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-11 and 13-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-11 and 13-20 have been examined.

Papers Received

2. Receipt is acknowledged of amendment papers submitted, where the papers have been placed of record in the file.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 1, 2, 4, 6, and 8 are rejected under 35 U.S.C. 102(b) as being anticipated by Levitan (EP 0605872 A1).

5. In regard to claim 1, Levitan discloses a method comprising:

a. Detecting a condition in a processor; Column 4, lines 56-58, show that certain instruction occurrences (conditions) control movement of data and thus are detected.

b. Calculating adjustment values at stages within a pipeline. Column 4, lines 55-56, shows that the system of mention is a pipelined processor. Column 6, lines 46-49, show that a register 46 receives values that are used for adjustment as shown in column 7, lines 22-31. Over a period of multiple cycles or stages multiple adjustment values are realized. At time N, a stage with an adjustment

value has been calculated. At time N+1, two adjustment values have been calculated over two different cycles and thus two stages.

c. Updating a register with one of the adjustment values when an instruction associated with the condition is terminated. Column 7, lines 22-31, show the case when the wrong sequence of instructions was taken after a conditional branch (BRANCH_ON_COUNT) causing an interrupt, or termination of the instruction. The section goes on to say that an adjustment value from register 46 is used to update register 42.

d. Wherein the stages within the pipeline comprise hardware stages, and data passes between the at least two of the stages during a processor cycle. [Column 2, line 10 of Levitan mentions that pipelined computers (with hardware stages) are used for the scope if the invention as does column 4, line 54 – column 5, line 1. Data inherently passes between pipeline stages in a processor. The examiner would like to note that the claim addition of a pipeline does not affect the other limitations as the new limitation is simply a statement of pipeline existence. Sine the processor in the reference is a pipelined processor there are stages inherent to it and the stages mentioned in the previous limitation do not need be the same stages.]

6. In regard to claim 2,

a. Levitan discloses the method of claim 1, as described above, wherein calculating the adjustment values comprises:

i. Incrementing the adjustment values when the condition is detected; Two detected instructions or conditions are the BRANCH_ON_COUNT and MOVE_TO_COUNT instructions. It is shown that the MOVE_TO_COUNT instruction is used only to load the update and dispatch registers with a count value (column 7, lines 5-11) so that the BRANCH_ON_COUNT instruction can be completed (column 7, lines 35-41). Thus when the processor detects that the MOVE_TO_COUNT is finished it is also detecting that the BRANCH_ON_COUNT instruction is about to begin. When it is detected that the BRANCH_ON_COUNT instruction is going to start, register 46, which holds the adjustment values, is updated with the count just loaded into register 44 (column 7, lines 35-38). Whether this is the first loop encountered or one in a series of many, the register 46 had held the value of zero indicating the end of the previous loop. Therefore, in updating the value of this register from zero to a number of iterations, the register is being incremented by that number of iterations.

ii. decrementing the adjustment values when the instruction leaves the stages. Column 7, lines 41-46, shows that the adjustment values held in register 46, as shown above, are decremented when the BRANCH_ON_COUNT is completed (thus leaving the current pipeline stages or cycles).

7. In regard to claim 4, Levitan discloses the method of claim 1, as described above, wherein detecting the condition comprises detecting an instruction within a hardware loop. The detected condition is a conditional branch (column 4, lines 56-58). Levitan shows in column 7, lines 2-3, that the branch instruction is that of a loop.

8. In regard to claim 6, Levitan discloses the method of claim 1, as described above, wherein detecting a condition comprises detecting a watch point. A watch point is set up to detect a condition such as a certain instruction as stated in the specification. The processor detects whenever a certain instructions are encountered as described above. In order to detect each instruction, it is inherent that a watch point must exist for it, and thus this watch point is detected for each instruction.

9. In regard to claim 8, Levitan discloses the method of claim 1, as described above, wherein updating the register comprises updating a speculative register. The register updated is register 42 as shown above. This register is a speculative register because the register (44) that controls its count (column 6, lines 49-53) is a speculative register because it corresponds to the count for an instruction speculatively executed as shown in column 6, lines 46-49, thus making register 42 also speculative.

10. Claims 9, 13, and 15 are rejected under 35 U.S.C. 102(b) as being anticipated by Gaertner (5,996,063).

11. In regard to claim 9, Gaertner discloses an apparatus comprising:

a. A first register; Column 13, lines 49-50, disclose two counters, R0/R1 and I0/I1. Column 13, lines 20-27, show that a rename instance is a register and thus, the R0/R1 counter, which is a rename instance, is also a register.

b. A second register; Column 11, lines 4-12, speak of a register file array where I0/I1 is a column in it and thus a register.

c. a set of counters to monitor the difference between the first register and the second register. It is well known that a set consists of one or more like elements. Column 13, lines 47-50, disclose a counter, or set of counters, U0/U1 that represents the difference between the two aforementioned counters, or registers.

d. Wherein the first register, second register and set of counters reside in a multi-stage pipeline controlled by a control unit, and the set of counters include counters maintained at a stage where the first register resides and at stages after the stage where the first register resides. It is inherent that a pipelined processor has a control unit to control it. The first register resides in the register renaming and allocation stage of figure 3 because the R0/R1 register is part of the allocator (figure 7). Since the first register keeps track of pointers to the rename registers, as shown above, it is accessed or updated whenever these registers are accessed. Over a period of multiple cycles or stages multiple counter values are realized and maintained. At time N, a stage with the counter is updated or maintained when the register is accessed. At time N+1, two counter values have been calculated over two different cycles or stages in the case that the register was accessed again. Thus a counter is maintained at cycle or stage N and cycle or stage N+1 where N+1 is after N and after this first stage where the first register resides.

e. Wherein the stages of the multi-stage pipeline comprise hardware stages, and data passes between at least two of the stages during a pipeline cycle. [Column 10, lines 14-23 shows that a pipelined processor is used. Data inherently passes between pipeline stages in a processor. The examiner would like to note that the claim addition of a pipeline does not affect the other limitations as the new limitation is simply a statement of pipeline existence. Sine the processor in the reference is a pipelined processor there are stages inherent to it and the stages mentioned in the previous limitation do not need be the same stages.]

12. Claim 10 is essentially the same as claim 9 with the exception that the first register is speculative and the second register is architectural. The second register is inherently architectural because it is within the architecture of the processor. The first register is speculative because it holds data regarding renaming instances as shown above. Renamed instances are not the actual registers but instead are another register used to work ahead on data instead of stall. In other words, the register is speculative because it provides for manipulation of data before the actual registers are ready and thus speculates.

13. Claim 11 states that the two registers are count registers, which has been previously shown.

14. In regard to claim 13, Gaertner has disclosed the apparatus as in claim 9, as described above, wherein the set of counters consist of counters residing in stages before an n^{th} stage of a pipeline, and wherein n defines a point at which allowing

instructions to flow through the pipeline takes an amount of time less than or equal to a branch penalty. It is well known in the art that a branch penalty is the number of cycles equal to the number of discarded instructions that are behind a branch instruction in the pipeline. The time to drain is taken to be the number of cycles from point n before the instructions thereafter are out of the pipeline. Figure 3 shows that based on the flow diagram, the register renaming and allocation block where the set of counters resides, as described above, is the fifth stage. Also from the diagram one can see that the most cycles performed after this point is 4 to the data memory. Therefore, if one defines n to be the 5th hardware pipeline stage, there are only at most 4 more hardware pipeline stages after this point for instructions to flow, which is less than or equal to the 4 that would need to be flushed before this point for a branch penalty.

15. In regard to claim 15, Gaertner discloses the apparatus as in claim 9, wherein the control unit is adapted to:

a. increment the counters when the first register is adjusted because of a detected condition; It is inherent that if the first register is adjusted it will be because of some detected condition. Because the set of counters U0/U1, is the difference between the first and second registers, as shown above, the counter is incremented if the first register is adjusted by a positive value such as in column 13, lines 39-40, where the first register is incremented.

b. decrement a respective counter when the instruction associated with the condition leaves a respective stage of the pipeline associated with the respective counter. In column 13, lines 37-40, it is shown how the register R0/R1 is

incremented when the register it points to is used as a target register. It is inherent that an instruction is using that register as a target. Now when that instruction is finished (or leaves a certain pipeline stage) the previous instance is no longer needed and thus is not the latest. Therefore, the first register, which keeps track of the latest rename instance as described before, will return to the previous position by decrementing. This will cause the set of counters U0/U1 to also decrement since it keeps a difference between the first and second registers. All of the hardware stages are *associated* with the respective counter since each hardware pipeline stage impacts the others and the counter is in a hardware pipeline stage

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

17. Claim 7 is rejected under 35 U.S.C. 103(a) as being unpatentable over Levitan.

18. In regard to claim 7,

a. Levitan discloses the method of claim 1, as described above, wherein updating the register with one of the adjustment values comprises adjusting the register by an amount determined by a counter as shown above, the register (42) is updated by register or counter 46 on an interrupt or termination.

b. Levitan does not specifically disclose that the counter is residing in the stage where the termination occurred.

c. One of ordinary skill in the art would have recognized that by rolling back the register to the non-speculative state in the same cycle or stage as the termination, no invalid data would advance and corrupt other registers. This ability to avoid data corruption would have motivated one of ordinary skill in the art to modify the design of Levitan to update the register in the same cycle or stage as the termination occurred.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Levitan to update the register in the same stage or cycle as the termination occurs so that no data corruption occurs from advanced invalid data.

19. Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable over Levitan in view of Gschwind (6,189,088).

20. In regard to claim 3,

a. Levitan discloses the method of claim 1, as described above.

b. Levitan does not disclose that detecting a condition comprises detecting an access to a specified memory location.

c. Gschwind has disclosed a speculative execution processor where in column 5, lines 41-55, the processor detects if a first load instruction loads a memory location read by a second, thus detecting an access to a specified memory location. Gschwind goes on to say that this interference must be detected so that the state of the processor can be recovered and the proper

instructions re-executed. This keeps the processor from processing invalid data. Incorporating this design into Levitan to also detect the condition of when a specified memory location is accessed would allow for the adjustment of the register when such a memory access condition was made in order to retrieve correct data.

d. The ability to detect interferences between instructions accessing a memory location would have motivated one of ordinary skill in the art to include the detection of an access to a particular memory location, as taught by Gschwind, into the design of Levitan.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Levitan to include the memory access detection of Gschwind in order to be able to detect memory interferences and avoid data errors.

21. Claim 5 is rejected under 35 U.S.C. 103(a) as being unpatentable over Levitan in view of Tran (6,003,128).

22. In regard to claim 5,

a. Levitan discloses the method of claim 4, as described above, of detecting a branch.

b. Levitan does not disclose wherein detecting the instruction within the hardware loop comprises detecting a bottom match. The specification has defined a bottom match to be the last instruction of a loop.

c. Tran has taught in column 1, lines 66-67, that a loop is delimited by the loop instruction, which executes as the last instruction in the loop. Tran defines the

loop instruction to consist of a decrement and a branch in the next few lines. Therefore the loop always ends in a branch.

d. By modifying the design of Levitan to be sure that loop iterations always end in the BRANCH_ON_COUNT instruction, the control would be simpler because it would always be known where the end of the loop was based on the detection of one instruction.

e. This simplified control would have motivated one of ordinary skill in the art at the time of invention to be sure that the design of Levitan called for loop iterations to always end with a branch as taught by Tran.

It would have been obvious to one of ordinary skill at the time of invention to modify the design of Levitan to always end loop iterations with a branch instruction so that control is simplified.

23. Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gaertner in view of Levitan.

24. In regard to claim 14,

a. Gaertner discloses the apparatus as in claim 12, as described above, wherein following a termination of an instruction in the pipeline, the control unit is adapted to adjust the first register. As shown in the summary, upon an exception, or termination, the reorder buffer and speculative registers are discarded and the in-order state is restored. This then includes the speculative first register of mention above.

b. Gaertner does not disclose that the first register is adjusted by an amount determined by a particular counter maintained in a stage where the termination occurred.

c. Levitan shows in figure 3, that register 42 is updated by counter 46 on an interrupt (termination). Column 6, lines 46-49 show that counter 46 is a counter because it holds a count value. This allows for the register to be restored immediately without having to fetch the correct data from memory or an architected register. The counter is maintained in each stage, including that where the termination occurs, so that the counter itself can be updated when needed, such as when it is decremented on completion of a branch (column 7, lines 41-46).

d. This quick restoration of the register to a correct stage would have motivated one of ordinary skill in the art to modify the design of Gaertner to include the adjustment method using a counter given by Levitan.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Gaertner to use the updating method for a speculative register using a counter as given by Levitan so that the updating may take place as quickly as possible and normal execution can resume.

25. Claims 10-11, 16-17, and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Levitan in view of Hennessy.

26. In regard to claim 16,

a. Levitan discloses

i. A memory device (figure 1, element 18);

ii. A first register (figure 3, element 44);

iii. A second register (figure 3, element 40);

iv. A set of counters; Figure 3, elements 42 and 46, are shown to hold a count value in column 6, lines 39-49, which is later decremented, thus yielding the elements to be counters.

v. A processor coupled to the memory device (figure 1), wherein the processor includes an execution pipeline (paragraph 2, line 2) and a control unit (column 7, lines 1-2) adapted to:

(1) Increment the counters when the first register is adjusted because of a detected condition; The detected condition is the MOVE_TO_COUNT instruction. It is shown that the MOVE_TO_COUNT instruction is used to load, or adjust, the update register, 44, with a count value (column 7, lines 5-11) so that the BRANCH_ON_COUNT instruction can be completed (column 7, lines 35-41). Thus when the processor detects that the MOVE_TO_COUNT is finished it is also detecting that the BRANCH_ON_COUNT instruction is about to begin. When it is detected that the BRANCH_ON_COUNT instruction is about to start, register 46, which holds the adjustment values, is updated with the count just loaded into register 44 (column 7, lines 35-38). Whether this is the first loop encountered or one in a series of

many, the register 46 had held the value of zero indicating the end of the previous loop. Therefore, in updating the value of this register from zero to a number of iterations, the register is being incremented by that number of iterations.

(2) Decrement a respective counter when the instruction leaves a respective stage of the pipeline associated with the respective counter. As stated previously, "the instruction" is taken to mean, "an instruction associated with the condition." Since the counter register architecture is initialized by the MOVE_TO_COUNT instruction so a BRANCH_ON_COUNT instruction can execute (column 7, lines 35-41), this branch instruction is associated with the detected move instruction, which is the condition. Column 7, lines 41-46, shows that the counter 46, is decremented when the BRANCH_ON_COUNT is completed (thus leaving the pipeline stages).

vi. Wherein the counters are maintained for corresponding hardware stages of the execution pipeline, and data passes between at least two of the hardware stages during a system cycle. [Column 2, line 10 of Levitan mentions that pipelined computers (with hardware stages) are used for the scope if the invention as does column 4, line 54 – column 5, line 1. Data inherently passes between pipeline stages in a processor. The examiner would like to note that the claim addition of a pipeline does not affect the

other limitations as the new limitation is simply a statement of pipeline existence. Sine the processor in the reference is a pipelined processor there are stages inherent to it and the stages mentioned in the previous limitation do not need be the same stages. The counters are indeed within corresponding pipeline stages.]

b. Levitan does not disclose that the memory device is a static random access memory device.

c. Hennessy discloses on page 541 in the table that SRAM or static random access memory is the fastest memory technology of the technologies disclosed.

d. The ability to access memory with an increased speed using the static random access variety as taught by Hennessy would have motivated one of ordinary skill in the art to modify the design of Levitan to use a static random access memory device.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Levitan to include a static random access memory device as disclosed by Hennessy in order to be able to access needed data quickly.

27. In regard to claim 17, Levitan discloses the system of claim 16, wherein following a termination of the pipeline, the control unit is adapted to adjust the first register. Levitan discloses the case of an interrupt, a pipeline termination, in column 7, lines 22-31. It is inherent then that the speculative register (column 6, lines 46-48), 44, must also be updated after such a termination in case of an incorrect value. This incorrect

value would exist if a MOVE_TO_COUNT had been executed before an interrupt that occurred before the branch was completed

28. In regard to claim 20, Levitan discloses the system of claim 17, as described above, wherein the control unit is adapted to drain unaborted instructions and write the first register with the data in the second register, if the termination occurs in a stage of the pipeline after an n^{th} stage. Column 5, lines 6-9, show that on a mispredicted branch (the interrupt or termination as stated above), the sequential instructions must be purged, or drained. Figure 3 shows that the first register, 44, is updated only by the second register, 40. Thus when the register is adjusted at a stage after an n^{th} stage, it is done so by writing the data in the second register to it.

29. Claims 18 and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Levitan in view of Hennessy as applied to claims 16-17 and 20 above, and further in view of Nakada (5,638,526).

30. In regard to claim 18,

a. Levitan discloses the system of claim 17, as described above, wherein the control unit is adapted to adjust the first register; As shown above, the first register is adjusted on a termination.

b. Levitan in view of Hennessy does not disclose that the first register is adjusted by an amount determined by one of the set of counters. This would consist of forwarding data from the counters instead of retrieveing from the register file or memory.

c. Nakada has disclosed a method of register bypassing in figure 1a where the needed data is retrieved from its first valid point where it can be selected rather than waiting for it to be written to and then read from the register. Anyone of ordinary skill in the art can se the performance advantage in this. The counters of Levitan in view of Hennessy will hold the correct data for adjustment when a branch has not yet completed. The first register will need this value if a MOVE_TO_COUNT has executed in the mean time before the exception. By adjusting the first register by an amount in the counters when possible, rather than waiting for an access to the second register, which is a general purpose register and thus in the register file, valuable time is saved.

d. This performance increase would have motivated one of ordinary skill in the art to use the register bypassing given by Nakada in the design of Levitan in view of Hennessy to forward data from the counters to the first register.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Levitan in view of Hennessy to include the register bypassing technique introduced by Nakada for forwarding data from the counters to the register file on an exception in order to decrease recovery time.

31. In regard to claim 19,

a. Levitan discloses the apparatus as in claim 12, as described above, wherein following a termination of an instruction in the pipeline, the control unit is adapted to adjust the first register where the termination occurred. Column 7, lines 41-46, shows that the adjustment values held in register 46, as shown

above, are decremented (by down counter 64 of figure 3) when the BRANCH_ON_COUNT is completed (thus leaving the pipeline and terminating). Because the adjustment happens upon termination, the two functions are synonymously linked and must happen in the same stage where such a branch termination can be detected.

b. Levitan in view of Hennessy does not disclose that the first register is adjusted by an amount determined by one of the set of counters. This would consist of forwarding data from the counters instead of retrieveing from the register file or memory.

c. Nakada has disclosed a method of register bypassing in figure 1a where the needed data is retrieved from its first valid point where it can be selected rather than waiting for it to be written to and then read from the register. Anyone of ordinary skill in the art can se the performance advantage in this. The counters of Levitan in view of Hennessy will hold the correct data for adjustment when a branch has not yet completed. The first register will need this value if a MOVE_TO_COUNT has executed in the mean time before the exception. By adjusting the first register by an amount in the counters when possible, rather than waiting for an access to the second register, which is a general purpose register and thus in the register file, valuable time is saved.

d. This performance increase would have motivated one of ordinary skill in the art to use the register bypassing given by Nakada in the design of Levitan in view of Hennessy to forward data from the counters to the first register.

It would have been obvious to one of ordinary skill in the art at the time of invention to modify the design of Levitan in view of Hennessy to include the register bypassing technique introduced by Nakada for forwarding data from the counters to the register file on an exception in order to decrease recovery time.

Response to Arguments

32. Applicant's arguments filed 8/12/04 have been fully considered but they are not persuasive.

33. Applicant has argued that the claims are now allowable because the references do not teach that the stages within the pipeline comprise hardware stages and that data passes between them. Column 2, line 10 of Levitan mentions that pipelined computers (with hardware stages) are used for the scope if the invention as does column 4, line 54 – column 5, line 1. Column 10, lines 14-23 of Gaertner show that a pipelined processor is used. Data inherently passes between pipeline stages in a processor Data inherently passes between pipeline stages in a processor. The examiner would like to note that the claim addition of a pipeline does not affect the other limitations as the new limitation is simply a statement of pipeline existence. Sine the processor in the reference is a pipelined processor there are stages inherent to it and the stages mentioned in the previous limitation do not need be the same stages.

Allowable Subject Matter

34. The examiner respectfully apologizes for incorrectly noting that claims 10 and 11 would be allowable if written in independent form. They are rejected as set forth above and as a result the Examiner making a non-final rejection.

Conclusion

35. The following is text cited from 37 CFR 1.111(c): In amending in reply to a rejection of claims in an application or patent under reexamination, the applicant or patent owner must clearly point out the patentable novelty which he or she thinks the claims present in view of the state of the art disclosed by the references cited or the objections made. The applicant or patent owner must also show how the amendments avoid such references or objections.

36. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The references cited in the previous Office Action remain pertinent and are cited herein.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane F Gerstl whose telephone number is (571) 272-4166. The examiner can normally be reached on M-F 6:45-4:15 (First Friday Off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Shane F Gerstl
Examiner
Art Unit 2183

SFG
November 1, 2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100